

# **EXHIBIT 4**



Sign in

Get started

Robinhood

ENGINEERING CULTURE IDEAS CRYPTO

# Under the Hood of Clearing by Robinhood

 Hon Hongxia Zhong [Follow](#)gxia Oct 16, 2018 · 5 min read  
Zhong

Last week, we announced Clearing by Robinhood — the first clearing system built on modern technology, from scratch, to directly integrate with some of the most critical parts of the trading industry. Since [Vanguard went self-clearing](#) in 2008, no other major brokerage has built anything similar in the past decade. This product has been the single most complex engineering and regulatory challenge we've undertaken as a company, and is a massive investment in the future of Robinhood. We're taking a moment from the late nights and engineering sprints to celebrate this two-year effort, the impact it will have for our customers, and the long awaited improvements it brings to the landscape of financial infrastructure.

755

Q 4



*Quick side note in case you're scratching your head wondering, "What is clearing?" (we're familiar with the feeling — most engineers came onto the project with no domain expertise, and that's fine!) — check out [this link](#) to get a clearer picture of what this product really is!*



Clearing Engineering and Product team at Robinhood

**Investing in Technology Upfront**

The road to this launch has been two years in the making and started with a small group of engineers learning and scoping the engineering requirements. It wasn't long before we had to make some challenging decisions, and one of them is to weigh the options of buying, building, or renting a system.

Building a clearing and settlement system in-house is a long term investment in Robinhood and our customers, as we become more vertically integrated in the financial system. We saw the opportunity to invest in modern engineering solutions in order to automate a traditional system and build on our own timelines. Investors, industry experts, and even brokerage veterans all shared the same warning: *nobody builds their own clearing system from scratch*, but they also convinced us that a more efficient clearing system would pass more savings to our customers and accelerate future product iterations. This directly aligns with our mission of **democratizing access to the financial markets**, so we knew we were on the right track.

Along our journey to self-clearing, we found many opportunities to invest up front in an automated system and reduce the long-tail cost. For example, many of our customers hold stocks that experience corporate actions (stock splits, reverse splits, mergers). Many clearing firms have a large number of operations specialists to manually support this process on a daily basis. After all, corporate action data is often unstructured and filled with edge cases. At Robinhood, two engineers built an entire system that enables this special processing for our millions of customers with only two brokers. This incredible degree of automation, created by our engineering and brokerage talent, is standard at Robinhood. (Later, we'll dive into how Jason, our Tech Lead for corporate actions, tackled this problem).

## A Modern Approach to Financial Infrastructure

You might be surprised to learn that some of the most critical financial systems were born out of something as rudimentary and catastrophic as *paperwork*. Back in the 1960s, when people called their broker to place a trade, the stock they bought or sold wasn't just a few pixels on their screen and a confirmation email in their inbox. An actual paper certificate had to pass through multiple hands to reach its new home. As investing became more commonplace and trading volumes began to spike, the army of messengers and clerks and brokers whose hands had to account for every physical trade began to fall behind. So far behind, in fact, that the stock market closed every Wednesday to give them a chance to catch up — leading to what the industry would soon call *the paperwork crisis*. Wall Street realized that only computers could handle the level of transactions they would continue to see and decided to overhaul the industry's approach to tracking stock ownership.



So in the 1960s, the financial industry made a heavy investment in automation and used the new boon of



Stock certificate — New Jersey Midland Railway Co., 1872

technology to solve this paperwork problem. Unfortunately, this technology investment has mostly remained unchanged since. Over 40% of banking systems still operate on COBOL, a language developed over 60 years ago when computers took up an entire room — and 92 of

the top 100 banks are still using mainframe computers. Broadly, the brokerage and clearing industries are no different. Until now.

In order to develop this large system at our own pace, we adopted the event-driven architecture pattern to achieve loose coupling between 20+ clearing modules affecting the ledger and the underlying accounting system. We also reaped the full benefits modern technologies like Python 3, Django/DRF, React JS, Kafka, Airflow, and more. Python and Django allowed us to model complex financial objects and quickly stand up API servers. We would have had a much harder time shipping nearly a hundred brokerage workflows without adopting modern web development frameworks like React JS and Redux.

An example of a DLQ on a Wall Street trading floor prior to the 1960s technology boon.

Further, Kafka offers a highly available queue with easy-to-manage delivery guarantees and glues together dozens of consumers and producers into a beautifully orchestrated event pipeline. In our stream-based approach to double entry accounting, we built tools to guarantee exactly once semantics and gracefully handle errors without deteriorating the integrity of our clearing system.

We run 300+ jobs every night to conclude the previous trading day and prepare for the next market open. Some financial companies coordinate these jobs by defining time-based dependencies with tools like Cron. Usually, these tools are hard to scale as the trading

volume and system complexity increases. At Robinhood, we use Airflow to build elastic dependencies between jobs and ensure that our system scales well with our rapid growth.

We're sprinting towards bringing this new system to all six million of our customers. With it, customers will benefit from significantly reduced ACH reversal fees and 24/7 application review and approval. What we've learned from this two-year effort will shape the next generation of Robinhood products. Stay tuned — many of them are in the works now. [Come join us](#) in building the next wave of industry-shaping financial products!

[Technology](#) [Fintech](#) [Engineering](#)


755 claps



4 responses



WRITTEN BY

**Hongxia Zhong**

Engineering Manager @ Robinhood

[Follow](#)**Robinhood**

Democratizing finance for all

[Follow](#)

## More From Medium

[Reflections on Robinhood's Growth](#)  
Jim Swartwout in Robinhood



[Breaking up the app-module monolith: the story of Robinhood's Android app](#)  
Jin Cao in Robinhood



[Android Inter-Module Navigation With Dagger](#)  
Jin Cao in Robinhood



[A Simple Implementation of Dynamic Configuration](#)  
Jae Hyeon Bae in Robinhood



[Remote Roundup—Meet Our Interns](#)  
Andrew Yeung in Robinhood



[An Introduction to Facebook's System Architecture: Social Graph and TAO](#)  
Meg's tech corner in The Startup



[A Pinterest Engineering guide to technical interviews](#)  
Pinterest Engineering in Pinterest Engineering Blog



[Digitalization of Production Facilities](#)  
International Society of Automation - ISA Official



### Learn more.

Medium is an open platform where 170 million readers come to find insightful and dynamic thinking. Here, expert and undiscovered voices alike dive into the heart of any topic and bring new ideas to the surface. [Learn more](#)

### Make Medium yours.

Follow the writers, publications, and topics that matter to you, and you'll see them on your homepage and in your inbox. [Explore](#)

### Share your thinking.

If you have a story to tell, knowledge to share, or a perspective to offer — welcome home. It's easy and free to post your thinking on any topic. [Write on Medium](#)